

# Projeto Ágil e Programação Eficaz

Carga Horária: 144 *Semestre 1º período*

## Ementa:

Identificação do problema real aplicando a metodologia design thinking; Representação de estrutura e comportamento de um modelo em diagramas UML: casos de uso, classe, sequência, colaboração e implantação; Programação Orientada a Objetos; Técnicas de Planejamento e Gerenciamento de Software com o framework ágil SCRUM; Documentação de projetos de Software; Estratégias para Construção de Software: TDD (desenvolvimento baseado em testes) e DDD (desenvolvimento baseado em domínio); Mapeamento de modelo de análise em modelo de objetos; Representação de estrutura e comportamento de um modelo em diagramas UML: casos de uso, classe, sequência, colaboração e implantação; Planejamento de implementação de um código; Requisitos funcionais e não funcionais de software; Critérios de qualidade de código; Estratégias de testes automatizados de software: testes de unidade; Técnicas eficientes de depuração de programas; Técnicas eficientes de explorar bases de código; Desenvolvimento em ambiente colaborativo e distribuído e controle de versão; Modelo de colaboração usando submissões, revisão de código, integração e entrega contínuos, testes automatizados e padrões de código; Sinalização e tratamento de erros e exceções; Monitoramento de software e arquivos de registro de histórico (log); Introdução à Engenharia de Software e fases do processo de desenvolvimento de software.

## Objetivos

Ao final da disciplina o aluno será capaz de:

1. Atuar em uma equipe que gere um projeto por meio do framework ágil SCRUM revezando papéis nesta equipe
2. Aplicar metodologia do design thinking para identificar e validar problemas de usuários
3. Identificar requisitos de sistema e expressá-los em casos de uso e histórias de usuário
4. Estruturar um sistema orientado a objeto a partir de requisitos
5. Expressar arquitetura e comportamento de um sistema a partir de diagramas UML
6. Utilizar ferramentas e soluções de desenvolvimento para que o time esteja apto a colaborar num projeto
7. Escrever testes de unidade que expressem comportamento esperado de um subsistema
8. Escrever documentação de software de forma organizada e clara
9. Analisar uma base de código pré-existente e propor intervenções
10. Resolver problemas de implementação a partir de resultados encontrados na Web e avaliar criticamente as fontes encontradas

11. Classificar requisitos como funcionais ou não-funcionais

### Conteúdo Programático

1. Design Thinking
2. Introdução à Engenharia de Software: fases do processo de desenvolvimento de software
3. Introdução ao framework ágil Scrum
4. Introdução à programação orientada a objetos
5. Diagramas UML
6. Introdução a UML
7. Mapeamento de requisitos em objetos de domínio
8. Como estruturar um projeto de software
9. Colaboração usando controle distribuído de versão
10. Desenvolvimento orientado a testes
11. Técnicas de depuração de código

### Bibliografia Básica

1. VALENTE, M.T. Engenharia de Software Moderna: Princípios e Práticas para Desenvolvimento de Software com Produtividade. Publicação Independente. 1.a edição. Belo Horizonte, 2020.
2. SCHWABER, K. , Agile Project Management with Scrum, Microsoft Press, 2004
3. THOMAS, D. HUNT, A. The Pragmatic Programmer: Your Journey to Mastery. 2.a ed. Addison-Wesley Professional, 2019.

### Bibliografia Complementar

1. CAROLI, P. Lean Inception: Como Alinhar Pessoas e Construir o Produto Certo. 1.a ed. Caroli, 2018.
2. MARTIN, ROBERT C. Clean Code: A Handbook of Agile Software Craftsmanship 2008. 1.a ed. Pearson.
3. MCCONNEL, S. Code Complete: A Practical Handbook of Software Construction. 2.a ed. Microsoft Press, 2004.
4. K.BECK., Test Driven Development: By example, Addison-Wesley Professional, 2002
5. PRESSMAN, R., Engenharia de Software: Uma abordagem profissional., 8ª ed., AMGH, 2016

## Sprint Session 2

Carga horária: 44h

### Ementa:

Elaboração da arquitetura de um software de armazenamento ou processamento de dados; Desenvolvimento de solução; Prototipação de software; Comunicação oral e escrita.

### Objetivos

Ao final da disciplina o aluno será capaz de:

1. Interpretar requisitos de um cliente relacionadas a um software centrado em dados
2. Propor uma solução
3. Esboçar a arquitetura de uma solução argumentando sobre restrições de projeto, demonstrando compreensão da computação em camadas e de forma integrada
4. Atuar em um time gerenciado por métodos ágeis na criação de um protótipo
5. Apresentar o projeto e criar documentação escrita

### Conteúdo programático

1. Coleta de requisitos de cliente
2. Arquitetura de software
3. Gestão de times ágeis
4. Comunicação escrita
5. Técnicas de apresentação

### Bibliografia Básica

WAZLAWICK, Raul. Análise e Design Orientados a Objetos para Sistemas de Informação. Rio de Janeiro: GEN LTC, 2014. E-book. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788595153653>. Acesso em: 31 de May 2022.

COHN, Mike. Desenvolvimento de Software com Scrum. Porto Alegre: Bookman, 2011. E-book. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788577808199>. Acesso em: 31 de May 2022.

DELAMARO, Marcio. Introdução ao Teste de Software. Rio de Janeiro: GEN LTC, 2016. E-book. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788595155732>. Acesso em: 31 de May 2022.

COHN, Mike. Desenvolvimento de Software com Scrum. Porto Alegre: Bookman, 2011. E-book. Disponível em:

<https://integrada.minhabiblioteca.com.br/#/books/9788577808199>. Acesso em: 31 de May 2022.

HIRAMA, Kechi. Engenharia de Software. Rio de Janeiro: GEN LTC, 2011. E-book. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788595155404>. Acesso em: 31 de May 2022.

URMA, Raoul-Gabriel; WARBURTON, Richard. Desenvolvimento Real De Software. Rio de Janeiro: Editora Alta Books, 2021. E-book. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9786555202021>. Acesso em: 31 de May 2022.

BES, Pablo; ALMEIDA, Claudinei de; SCHOLZ, Robinson Henrique et al. Soft Skills. Porto Alegre: SAGAH, 2021. E-book. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9786556901244>. Acesso em: 31 de May 2022.

MACIEL, Francisco Marcelo de Barros. Python e Django. Rio de Janeiro: Editora Alta Books, 2020. E-book. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9786555200973>. Acesso em: 31 de May 2022.

1. MARTIN, ROBERT C. Clean Code: A Handbook of Agile Software Craftsmanship 2008. 1.a ed. Pearson.
2. THOMAS, D. HUNT, A. The Pragmatic Programmer: Your Journey to Mastery. 2.a ed. Addison-Wesley Professional, 2019.
3. VALENTE, M.T. Engenharia de Software Moderna: Princípios e Práticas para Desenvolvimento de Software com Produtividade. Publicação Independente. 1.a edição. Belo Horizonte, 2020.
4. BROWN, TIM. Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation, 2019.

### Bibliografia Complementar

4. MCCONNEL, S. Code Complete: A Practical Handbook of Software Construction. 2.a ed. Microsoft Press, 2004.
5. SOMMERVILLE, I. , Engenharia de Software , 9ª ed., Pearson, 2011
6. SCHWABER, K. , Agile Project Management with Scrum, Microsoft Press, 2004
7. PRESSMAN, R., Engenharia de Software: Uma abordagem profissional., 8ª ed., AMGH, 2016
8. K.BECK., Test Driven Development: By example, Addison-Wesley Professional, 2002
9. BECK, K.; ANDRES, C. , Extreme programming explained: Embrace change, Addison-Wesley,, 2005